UNIQUIMER 3D, *A software system for structural DNA nanotechnology design, analysis and evaluation*

Manual for version 1.0.

User's Guide to the UNIQUIMER 3D

Version 1.0

http://ihome.ust.hk/

Jinhao Zhu, Yongli Mi
Email: csnickle@gmail.com, keymix@ust.hk

December 30, 2008

Contents

I Introduction **4**

II Getting Started **4**

III Building a Motif **4**

IV Geometrical Optimization **7**

V Sequence Generation **10**

VI Summary **12**

Part I
Introduction

A user-friendly software system, *UNIQUIMER 3D*, was developed for structural DNA nanotechnology that consists of 3D visualization, internal energy minimization, sequence generation, and construction of motif array simulations (2D tiles & 3D lattices) functionalities. The system can be used to check the structural deformation and design errors under scaled-up conditions. *UNIQUIMER 3D* aims to facilitate the design of novel DNA motifs for the DNA nanotechnology and has been tested for the design of both existing motifs (holiday junction, 4x4 tile, DX, DNA tetrahedron, DNA cube, etc.) and not-existing motif (soccer ball). A *de novel* sequence generation algorithm is integrated into the system. *UNIQUIMER 3D* was developed for the Windows environment and is provided free of charge to the non-profit research institutions.

In this user manual documentation, we start with an explanation of how to install *UNIQUIMER 3D*. Next, a short tutorial is given that explains most of the features that you'll need in designing SDN. Following the tutorial you will find a "possible workflow" for creating a DNA motif. Following this workflow may help you avoid problems later on. After that, we will describe how to make use of the energy minimization functionality to optimize the geometrical shape of a designed structure. Then we will demonstrate how to apply sequence generation algorithm to your design. At the end of the documentation you will find a summary of our work.

Part II
Getting Started

This part helps you getting started. It starts with an explanation of how to install *UNIQUIMER 3D*. *UNIQUIMER 3D* can be accessed from `http://ihome.ust.hk/` for free. The first step is to grab a copy of the UNIQUIMER 3D (zip archive), then uncompress the file to your local hard drive. The folder "Structures" contains a collection of some default existed motifs, such as *holiday junction, 4x4 tile, cube, tetrahedron, etc.* The folder "Report" holds a collection of all the HTML reports you generate for a designed structure. You will also find the executable file, namely "UNIMQUIMER3D.exe" in the unzipped parent directory. The major user interface of *UNIQUIMER 3D* is shown in Fig 1.

There is a "Item List" tree control on the left docking window. All the SDN information are maintained in this control as a tree structure. A "Information Window" is docked on the left bottom side that displays the detailed information of the user selections. Users can insert, translate, rotate items by making use of the "Control Panel" docked on the right bottom side of the window. The center window is the 3D display canvas.

Part III
Building a Motif

In *UNIQUIMER 3D*, DNA strands can be manipulated on a 3D canvas (addition, deletion, translation, rotation, etc.) Basic DNA components can be joined together at crossover points to build certain motifs.

This part contains an explanation of all the steps that are necessary to create a DNA motif. The basic building block is the DNA double helical domain.

The most basic element and the lowest level of our system hierarchy is the DNA base pair. Each DNA base pair consists of a pair of nucleotide residues (DNA nodes hereinafter) connected by hydrogen bonds, and DNA nodes are rendered as spheres. Phosphodiester linkages and hydrogen bonds are rendered as lines of different colors in the 3D space. Individual nucleotide residues (DNA customized nodes hereinafter) are also modeled and rendered as spheres in our system.

The double helical domain contains a parent node rendered as box, a green strand and a blue strand as shown in Fig 3. The polarity of the double helical domain is defined. Starting from the parent node, the green strand is pointing upwards all the way to the topmost base and the blue strand then pointing downwards till reaching the parent node. The parent node determines the position of the double helix. The direction of double

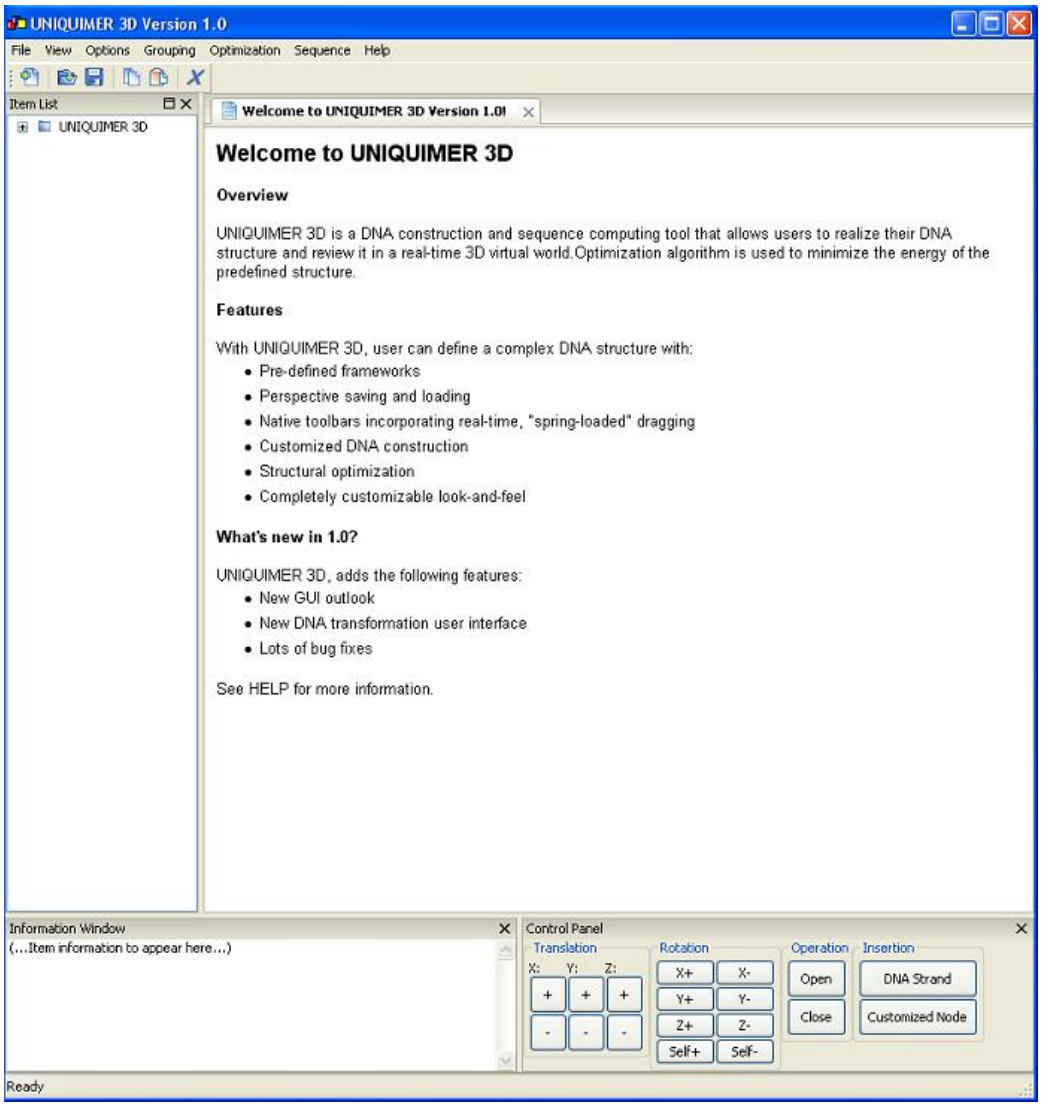
UNIQUIMER 3D Version 1.0
File View Options Grouping Optimization Sequence Help
Item List
UNIQUIMER 3D
Welcome to UNIQUIMER 3D Version 1.0!
Welcome to UNIQUIMER 3D
Overview
UNIQUIMER 3D is a DNA construction and sequence computing tool that allows users to realize their DNA structure and review it in a real-time 3D virtual world.Optimization algorithm is used to minimize the energy of the predefined structure.
Features
With UNIQUIMER 3D, user can define a complex DNA structure with:
Pre-defined frameworks
Perspective saving and loading
Native toolbars incorporating real-time, "spring-loaded" dragging
Customized DNA construction
Structural optimization
Completely customizable look-and-feel
What's new in 1.0?
UNIQUIMER 3D, adds the following features:
New GUI outlook
New DNA transformation user interface
Lots of bug fixes
See HELP for more information.
Information Window
(...Item information to appear here...)
Control Panel
Translation
Rotation
Operation
Insertion
X: Y: Z:
X+ X- Y+ Y- Z+ Z- Self+ Self-
Open
Close
DNA Strand
Customized Node
Ready

Figure 1: System User Interface.

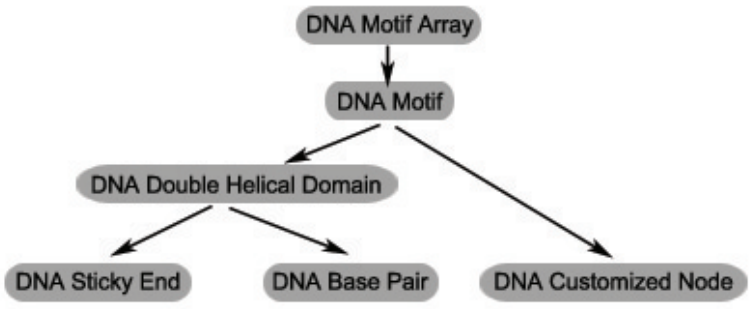

Figure 2: System Hierarchy.

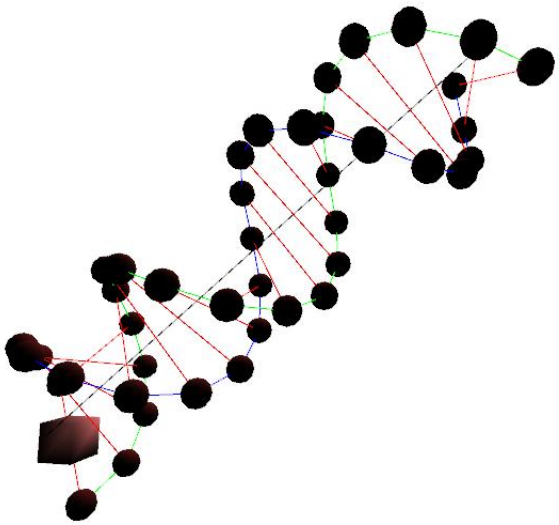

Figure 3: DNA Double Helical Domain.

helix is defined to be a ray casting from the parent node to the center position of the topmost base pair along the center pole.

A double helical domain can be added to the 3D canvas by clicking on the "DNA Strand" button in the "Control Panel" as shown in Fig 4.

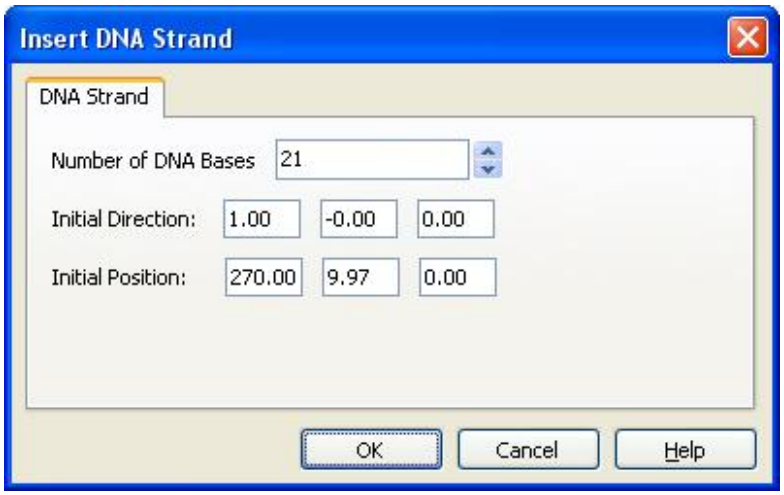

Figure 4: DNA Double Helical Domain Insertion.

The top textbox allows the user to input the number of base pairs of the DNA double helix. The middle textbox is for inputting the initial direction of the DNA double helix. The bottom textbox is for inputting the initial position of the DNA double helix.

A grouping operation is embedded in our system so that multiple DNA double helices with DNA customized nodes can be regrouped for further manipulation or management. The grouped components are maintained in a tree structure (parent-children structure) that is consistent with the system hierarchy.

Two more operations are defined for DNA constructions: (1) the opening operation that breaks the con-

secutive DNA nodes; (2) the closing operation that connects two open nodes. The two operations redefine connectivity between DNA nodes. Invalid polarities are automatically detected by the system and will not be allowed (as illustrated in Fig 5).

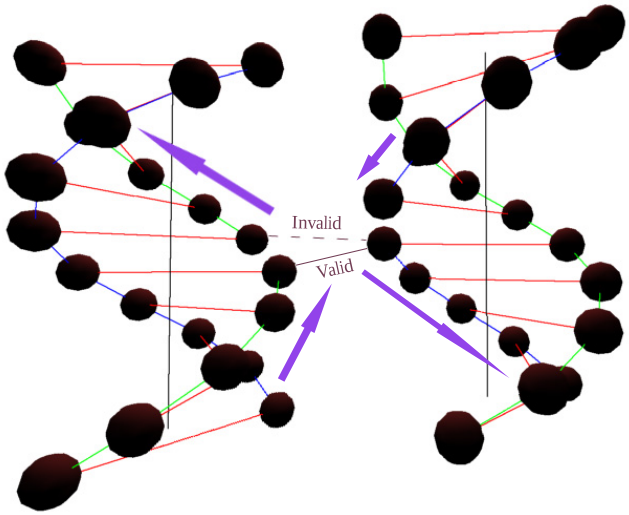

Figure 5: Valid and invalid closing operations.

Part IV
Geometrical Optimization

offers ways to change the geometrical appearance of a designed structure. Given the user-defined structure (hierarchy and connectivity), a state is defined to be the entire geometric information of all of its components. An energy function is designed to assign a nonnegative real number representing its stability to each state.

Energy function is introduced to eliminate the structural defects and design errors, which might result in constructional failure, thus to get a better evaluation whether or not a specific DNA structure is able to form in a stable way. For random configurations of DNA strands, the energy function may be very complicated. However, the regular and predictable double helical DNA structure makes the energy minimization relatively simple. Accordingly, the distance between two nucleotide residues and the smoothness of the double helix are taken into consideration. So the energy function is defined as

$$\mathbf{E} = (1-\lambda)\mathbf{E}_{distance} + \lambda\mathbf{E}_{smoothness}, \tag{1}$$

where $\mathbf{E}_{distance}$ and $\mathbf{E}_{smoothness}$ are two terms that are consistent with our motivation, and λ is a weight.

Since motif arrays are assembled by motifs, and motifs are further composed of DNA double helices, sticky ends and DNA customized nodes, on the basic level, only geometric information for DNA double helices, sticky ends, DNA customized nodes and the corresponding connectivity is necessary to model our energy function.

Given a set of n DNA double helical domains and their corresponding rotation and translation parameters, Θ_is and T_is, and m DNA customized nodes with translation vectors, L_js, a state can be defined as (Θ, Γ), where $\Theta = (\Theta_1, \Theta_2, \ldots, \Theta_n)$, $\Gamma = (T_1, T_2, \ldots, T_n, L_1, L_2, \ldots, L_m)$.

$\mathbf{E}$ is a function mapping from (Θ, Γ) to nonnegative real numbers.

$\mathbf{E}_{distance}$ is defined as the sum of squared differences of the distance between each pair of connected nodes and a constant, d. d is chosen to be the distance between any two consecutive DNA nodes on the same helix, which is 63.05. Therefore, with a user-defined connectivity map, $\mathcal{C} = \{< \sigma_1, \rho_1 >, < \sigma_2, \rho_2 >, \ldots, < \sigma_k, \rho_k >\}$,

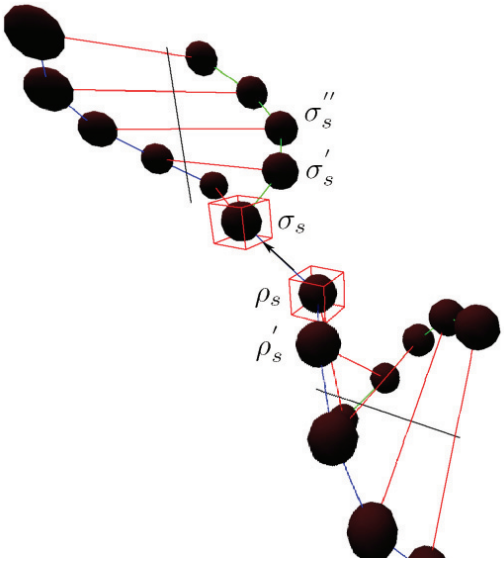

Figure 6: Illustration of the smoothness term.

where σ_s and ρ_s ($s = \{1, 2, \ldots, k\}$) denote user connected nodes and $< \cdot >$ denotes connectivity.

$$\mathbf{E}_{distance} = \sum_{s=1}^{k} (\|P(\sigma_s) - P(\rho_s)\| - d)^2, \tag{2}$$

where $\|\cdot\|$ is the Euclidean norm and $P(\cdot)$ is a function that denotes the position of a node. It should be noted that σ_s (ρ_s) is either a node on a helix or a DNA customized node. If the node is on a helix, it will have position $\mathcal{R}(\Theta_i) \cdot h^*(t) + T_i$, where Θ_i, t, T_i and $h^* = \{h^+, h^-\}$ depend on the working structure. If the node is a DNA customized node, j, its position is simply L_j, depending on the structure.

The second term, $\mathbf{E}_{smoothness}$, in our energy function refines the smoothness of the user connection of DNA double-helical domains.

In an ideal case of DNA strand connection, the B-form DNA's helical structures are preserved. As shown in Fig 6, two DNA double helices are joined together by connecting nodes, σ_s and ρ_s. An optimal angle between the two vectors formed by any three neighboring DNA nodes on the same helix, $\sigma_s, \sigma_s', \sigma_s''$, is defined to be $\theta_{optimal} = \arccos\left(\frac{\overrightarrow{\sigma_s\sigma_s'}}{\|\overrightarrow{\sigma_s\sigma_s'}\|} \cdot \frac{\overrightarrow{\sigma_s'\sigma_s''}}{\|\overrightarrow{\sigma_s'\sigma_s''}\|}\right)$. $\theta_{optimal}$ is a constant that equals to 24.53. It is desirable to have a connection $\overrightarrow{\rho_s\sigma_s}$ that forms the same angle, $\theta_{optimal}$, with both $\overrightarrow{\sigma_s\sigma_s'}$ and $\overrightarrow{\rho_s'\rho_s}$. Therefore, the smoothness term is defined as

$$\begin{aligned}\mathbf{E}_{smoothness} = & \sum_{\sigma_s, \rho_s} \left(\left(\arccos\left(\frac{\overrightarrow{\sigma_s\sigma_s'}}{\|\overrightarrow{\sigma_s\sigma_s'}\|} \cdot \frac{\overrightarrow{\sigma_s\rho_s}}{\|\overrightarrow{\sigma_s\rho_s}\|}\right) - \theta_{optimal}\right)^2\right. \\ & \left. + \left(\arccos\left(\frac{\overrightarrow{\rho_s\rho_s'}}{\|\overrightarrow{\rho_s\rho_s'}\|} \cdot \frac{\overrightarrow{\sigma_s\rho_s}}{\|\overrightarrow{\sigma_s\rho_s}\|}\right) - \theta_{optimal}\right)^2\right).\end{aligned} \tag{3}$$

The summation is over all connections between nodes σ_s and ρ_s on the helices. σ_s' denotes the node next to σ_s, which is not ρ_s; σ_s'' denotes the node next to σ_s', which is not σ_s; ρ_s' denotes the node next to ρ_s, which is not σ_s. This smoothness term penalizes angular discrepancies from $\theta_{optimal}$ of the connected structure.

Our energy function takes (Θ, Γ) as variables. Given an initial user-defined state, (Θ, Γ), we want to improve it using an energy minimizing technique. Currently the properties are controlled using simple geometry as defined in $E_{distance}$ and $E_{smoothness}$, which is a convex function. As a result, gradient-based local minimization algorithm is considered in this version. Powell's method, which is an iterative optimization method that finds a local optimizer, is implemented in our system for this purpose.

Since the result from running Powell's method greatly depends on its initial state, it is very important to supply a relatively stable state with low energy as the input to Powell's method. We first coarsely scan through states that uniformly cover the solution space (Θ, Γ) to find a state with the lowest energy. Since $\mathbf{E}$ is a smooth function, it is reasonable to assume that a global minimum exists somewhere near this state. Therefore, this selected state is the starting point of the Powell's method.

One should note that there is no guarantee that the final state, which is a local minimum, is a global one. However, if the initial state is close enough to the global minimum, it can be found using Powell's method.

As shown in Fig 7(a) and Fig 7(c), certain DNA double-helical domains are set to be distorted. After applying energy minimization to the structures, the geometrical shapes of these double-helical domains are refined as shown in Fig 7(b) and Fig 7(d).

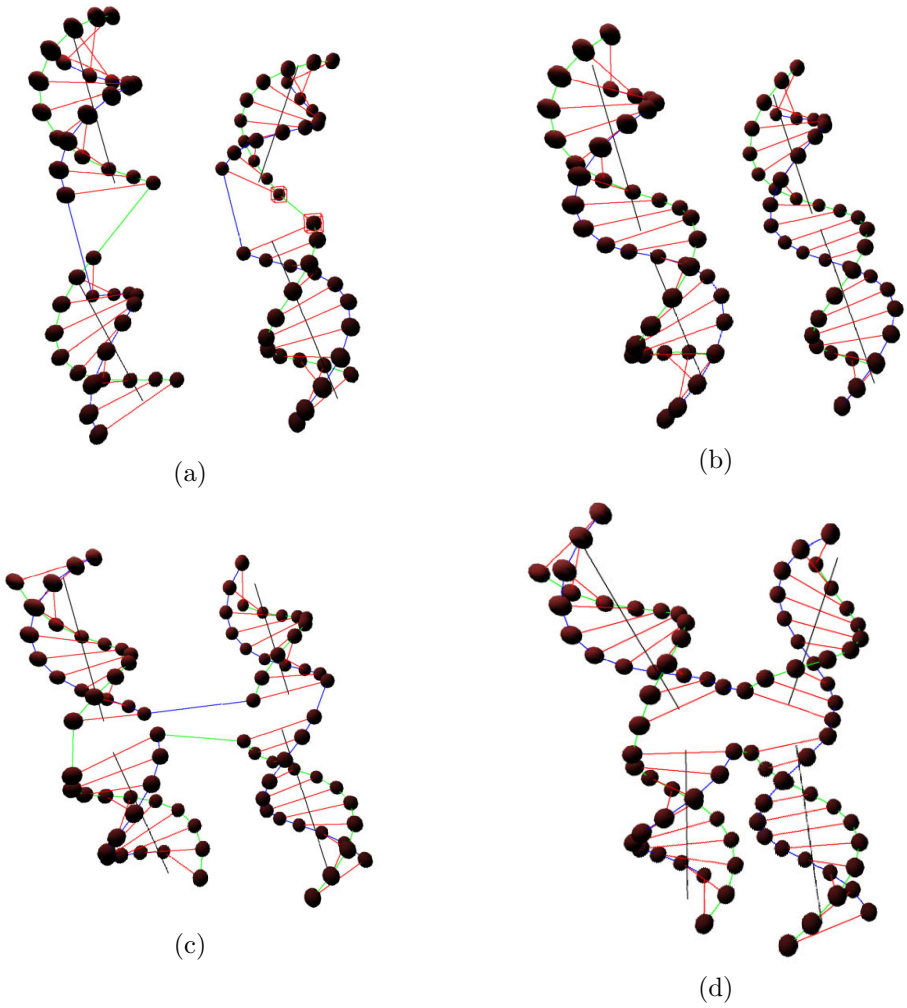

Figure 7: Geometrical Optimization. (a) Two distorted DNA double-helical domains before energy minimization. (b) Two refined DNA double-helical domains after energy minimization. (c) Two distorted DNA double-helical domains with crossover before energy minimization. (d) Two refined DNA double-helical domains with crossover after energy minimization.

After the user has selected the portion needed to be optimized, a energy minimization dialog is shown by selecting the "Optimization" in the menubar. The dialog contains three textbox inputs, allowing the user to specify the number of iterations the optimization algorithm will do, the translation and rotation offset in the 3D space, and five selection boxes, which allows the user to disable/enable translation, global/local rotation or collision detection. The dialog is shown in Fig 8.

UNIQUIMER 3D will generate a detailed report in HTML format of the refined structure including information about its hierarchy and several showcase images of the structure from different viewing angles. Besides, the process of energy minimization is illustrated in this report with a chart showing each iteration in this process

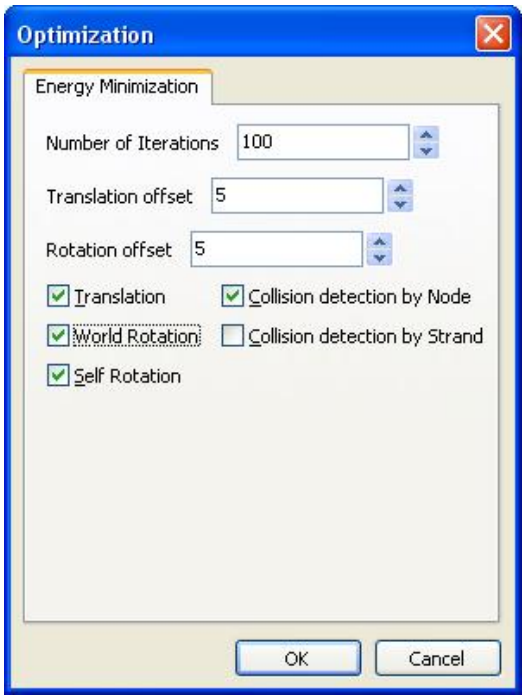

Figure 8: Geometrical Optimization Dialog.

and the corresponding energy value.

Part V
Sequence Generation

A sequence generation algorithm is designed and integrated it to *UNIQUIMER 3D*. The algorithm of generating sequence sacrifices storage to gains speed. The idea is to compute all the possible combinations of the specific maximum length of repetitive segment starting with {A, T, G, C}. The combination term is an unordered collection of distinct elements, usually of a prescribed size and taken from a given set. This approach guarantees each of the segment is distinct in terms of sequence so that the length of repetitive segment is controlled. The combinations can be pre-computed and stored to a local file that can be loaded for recycling usage.

The same basic rules for SDN sequence generation are taken into consideration for *UNIQUIMER 3D*. The first one to follow is the pairing up rule of {A=T}, {G≡C} (i.e. certain segments should be complementary respectively as shown in Fig 9).

In order to avoid segment mismatching cases as much as possible, there is the second rule, to limit the length of repetitive segments. It is illustrated in Fig 10. If the requirement is set to have no repetitive segments of 4 base pairs, the sequence does not meet the requirement. However, if the requirement is set to have no repetitive segments of 5 base pairs, the sequence will pass. As the main restriction used for sequence generation, the maximum length of repetitive segment should be set as short as possible to prevent mismatching cases. If the value is set to be 3, there will be no repetitive segment with a length of 4 or more bases. Suppose we want to do sequence generation for a structure of a DNA double helical domain with 100 base pairs. The total number of combinations for segments with a length of 3 bases is only $4^3 = 64$, so if the maximum length of repetitive segment is set to be 3, there would be not enough candidates to fill in the 98 (100-3+1) blanks of segments with a length of 3 bases. Therefore, no solution could be found in this case. However, if the value is set to be 4 instead, the possible combinations increase to $4^4 = 256$. There are enough candidates available in this case and the generator will find a solution. In general, the maximum length of repetitive segment is relatively bigger

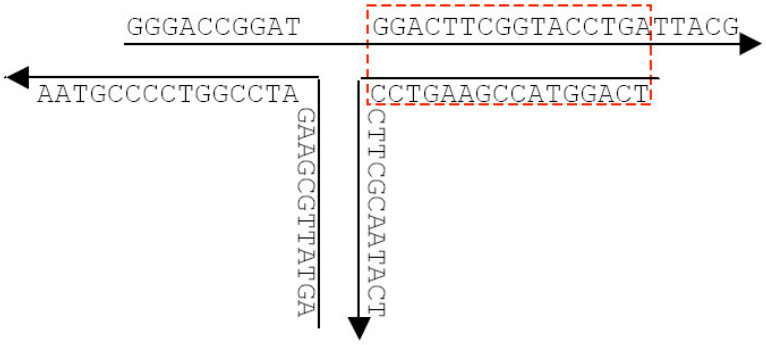

Figure 9: Base pair matching. Sequences shown in the box are subjected to complementary region.

for complicated structures compared with simpler structures like DX or TX. The value ranges from 4 to 6 for most of the SDN structures. However, it could be as big as 7 or more for extremely complicated structures. The maximum length of the repetitive segment in the sticky ends is set to be 3 base pairs (no repetitive segment of 4 base pairs) no matter what the global rule of repetitive segment length minimization is.

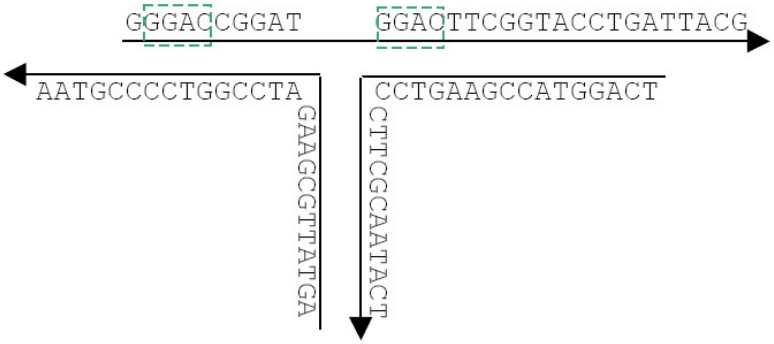

Figure 10: Mismatching prevention. Sequnces shown in the box are two repetitive segments of 4 base pairs.

UNIQUIMER 3D is able calculate the number of potential mismatching cases that are hindrances of formation of the desired structure. Given the length of potential mismatching segment, a scoring system of a generated sequence for a structure is formulated as following:

Pair is defined as

$$Pair(a,b) = \begin{cases} 1 & [a,b] \in \{[\text{A,T}],[\text{T,A}],[\text{G,C}],[\text{C,G}]\} \\ 0 & otherwise \end{cases} .$$

Comp is defined as

$$Comp(a,b) = \begin{cases} 1 & a = b \\ 0 & otherwise \end{cases} .$$

$\mathscr{D} = \{D_1, D_2, \ldots, D_n\}$ is the set of DNA double helical domains, $D_i \in \mathscr{D}$ is a particular double helical domain with $|D_i|$ base pairs $\{< \mathscr{N}_i^1, \overline{\mathscr{N}_i^1} >, < \mathscr{N}_i^2, \overline{\mathscr{N}_i^2} >, \ldots, < \mathscr{N}_i^{|D_i|}, \overline{\mathscr{N}_i^{|D_i|}} >\}$. given the digit d that represents for the length of potential mismatching segment, a score function is defined as

$$J = \sum_{i=0}^{|\mathscr{D}|} \sum_{j=0}^{|D_i|-d} Comp[\sum_{k=0}^{d} Pair(\mathscr{N}_i^{j+k}, \overline{\mathscr{N}_i^{j+k}}), d]$$

The score function J is a numerical representation of the expected formation of the structure consisting of a set of DNA double helical domains $\mathscr{D}$. In real self-assembly process, potential mismatching cases will result in secondary structure formation. $\mathscr{S} = \{S_1, S_2, \ldots, S_n\}$ is the set of DNA strands, $S_i \in \mathscr{S}$ is a particular strand with $|S_i|$ bases $\{\mathscr{N}_i^1, \mathscr{N}_i^2, \ldots, \mathscr{N}_i^{|S_i|}\}$. We then define another score function $\hat{J}$ taking into account all potential mismatching cases as a comparison with the value we get from J. Since J is the most optimal score, $\hat{J}$ is always greater than or equal to J. The difference between these $\hat{J}$ and J indicates how likely the strands $\mathscr{S}$ with generated sequences will self-assemble into the expected structure. The less difference between the values we get from $\hat{J}$ and J, the more likely $\mathscr{S}$ will self-assemble into the expected structure formation.

$$\hat{J} = \sum_{i=0}^{|\mathscr{S}|} \sum_{p=0}^{|S_i|-d} \sum_{j=0}^{|\mathscr{S}|} \sum_{q=0}^{|S_j|-d} Comp[\sum_{k=0}^{d} Pair(N_i^{p+k}, N_j^{q+k}), d]$$

$\hat{J}$ is designed to find all potential mismatching cases, which is equivalent to string matching in terms of conjugating one of the segment to search for all matchings of this conjugation from a given set of segments. The conjugate operation is based on DNA complementarity {A=T}, {G≡C}, {T=A}, {C≡G}. Consider segment $[N_i^p, N_i^{p+1}, \ldots, N_i^{p+k}]$ as the template, it is equivalent to essentially searching on strand j for the total number of conjugate segments. As a result, Boyer-Moore string search algorithm is adopted for $\hat{J}$.

Our score functions take a structure with sequence and the length of potential mismatching segment d as input. A set of sequence generated for n times of a structure is denoted as $\Phi = \{\Phi_1, \Phi_2, \ldots, \Phi_n\}$. The corresponding numbers of potential mismatching cases are denoted as $\Omega = \{\hat{J}_1 - J_1, \hat{J}_2 - J_2, \ldots, \hat{J}_n - J_n\}$. A tuple $\Delta^d = [\Omega, \Phi]$ is associated. $\Omega_i = min(\Omega)$, $\Delta_i^d = [\Omega_i, \Phi_i]$ is selected for a specific d.

The sequence generation dialog is shown is Fig 11. The user has options to specify the maximum length of repetitive segment and a exclusion set. The result of the sequence generation is saved to a text file in the parent directory namely "strandSequence_All.txt". It contains all the sequences of each individual strand of the structure.

Figure 11: Sequence Generation Dialog.

Part VI
Summary

With all these built-in features, users can easily design, analyze and evaluate different structures. Especially, the energy minimization function can help users to obtain a relatively stable state of the working structure and the result can be very helpful for SDN prediction. DNA structures without satisfactory optimized state can be screened out for wet-lab experiments.

The main contributions of this work can be summarized as follows:

- Users can visualize DNA motifs and motif assemblies in a 3D environment.

- Users can design DNA structures in a convenient and efficient way.
- An energy function is designed for measuring the stability of structures. Our system can relax this energy function to predict a relatively stable structure, which can validate and/or predict SDN wet-lab experiments.
- Each DNA node in a structure can be automatically assigned a tag from {A, T, G, C} using a built-in sequence-generating algorithm, and the generated sequence can be analyzed by our scoring system.
- A detailed HTML report is generated after the energy minimization, which contains hierarchical information on the refined structure, showcases images of it from different viewing angles and gives information on the energy minimization.

As for future development, a systematic analysis of our current energy minimization will be carried out down to the molecular level of DNA backbone structures. The current single-level algorithm makes the efficiency of the minimization process low. Multi-level of optimization which minimize all factors simultaneously at each iteration will be utilized to address this issue. We will consider the simulated annealing method that can sample a wider range of conformations compared with other gradient based local minimization methods. We will also work on the modeling down to the molecular level with precise atomic positional control since rotations of each bond in the backbone structure will result in different conformations of the entire design and the construction using different forms of DNA or even RNA. This will be updated into our next release and will enable controls of the base and sugar. It will be much more accurate compared with the current version that models only simplified DNA. PDB format of exportation will also be supported by then.